**EXPERIMENT NO. 7**

**TO DETECT EDGES IN THE IMAGE WITH THE HELP OF PREWITT’S AND SOBEL’S EDGE DETECTORS AND COMPARE THEIR PERFORMANCES**

**EXPERIMENT NO.:7**

**AIM: -** To detect edges in the image with the help of Prewitt’s and Sobel’s edge detectors and compare their performances.

**OBJECTIVE:**

1. Tounderstand concept of edge detection operators.

**EQUIPMENTS/SOFTWARE: Scilab6.0.0**

**THEORY: -**

Technically speaking, image segmentation refers to the decomposition of a scene into different components (thus to facilitate the task at higher levels such as object detection and recognition)

Segmentation subdivides an image to regions or objects

Two basic properties of intensity values

* Discontinuity

– Edge detection

* Similarity

– Thresholding

– Region growing/splitting/merging

**Detection of Discontinuities: Point Detection**

**Mask operation**

• Point detection  

Isolated point  whose gray value is significantly different from its background



**Edge Detection: Gradient Operators**

Gradient

– Vector pointing to the direction of maximum rate of change of *f* at coordinates (*x*,*y*)

– Magnitude: gives the quantity of the increase (sometimes referred to as *gradient* too)

– Direction: perpendicular to the direction of the edge at (*x*,*y*)

– Partial derivatives computed through 2x2 or 3x3 masks

•Sobel operators introduce some smoothing and give more importance to the center point



 (Magnitude)  (Direction)  

**Detecting diagonal edges**

****

**ALGORITHM**

1. Read input image
2. Define mask for particular operator
3. Move mask over image and calculate horizontal and vertical edges from the image
4. Display all detected edges.

**FUCTIONS**

1. **imread()**
2. **imshow()**
3. **double()**

**PROGRAM:**

*//EXP 7*

*//Dikshita Kambri 118A2044*

clc;

clear all;

im=imread("C:\Users\hp\Documents\Image Processing-Scilab\Images\toyobjects.png");

[r c] = size(im);

im = double(im);

prex = (1/6)\*[-1 -1 -1;0 0 0;1 1 1];

prey = (1/6)\*[-1 0 1;-1 0 1;-1 0 1];

sobx = (1/8)\*[-1 -2 -1;0 0 0;1 2 1];

soby = (1/8)\*[-1 0 1;-2 0 2;-1 0 1];

*//Z=zeros(r+2,c+2);*

*//Z(2:r+1,2:c+1)=im;*

outprex=zeros(r,c);

outprey=zeros(r,c);

outsobx=zeros(r,c);

outsoby=zeros(r,c);

for i=2:r-1

for j=2:c-1

x1 = im(i-1:i+1, j-1:j+1);

outprex(i,j) = sum(prex.\*x1)

outprey(i,j) = sum(prey.\*x1)

outsobx(i,j) = sum(sobx.\*x1)

outsoby(i,j) = sum(soby.\*x1)

end

end

prewitt = abs(outprex) + abs(outprey);

sobel = abs(outsobx) + abs(outsoby)

figure(1)

imshow(uint8(im));

title('Original Image');

figure(2)

subplot(1,3,1),

imshow(uint8(abs(255\*outprex/max(max(outprex)))));

title('Prewitt X-gradient');

subplot(1,3,2),

imshow(uint8(abs(255\*outprey/max(max(outprex)))));

title('Prewitt Y-gradient');

subplot(1,3,3),

imshow(uint8(255\*prewitt/max(max(prewitt))));

title('Prewitt Final');

figure(3)

subplot(1,3,1),

imshow(uint8(abs(255\*outsobx/max(max(outprex)))));

title('Sobel X-gradient');

subplot(1,3,2),

imshow(uint8(abs(255\*outsoby/max(max(outprex)))));

title('Sobel Y-gradient');

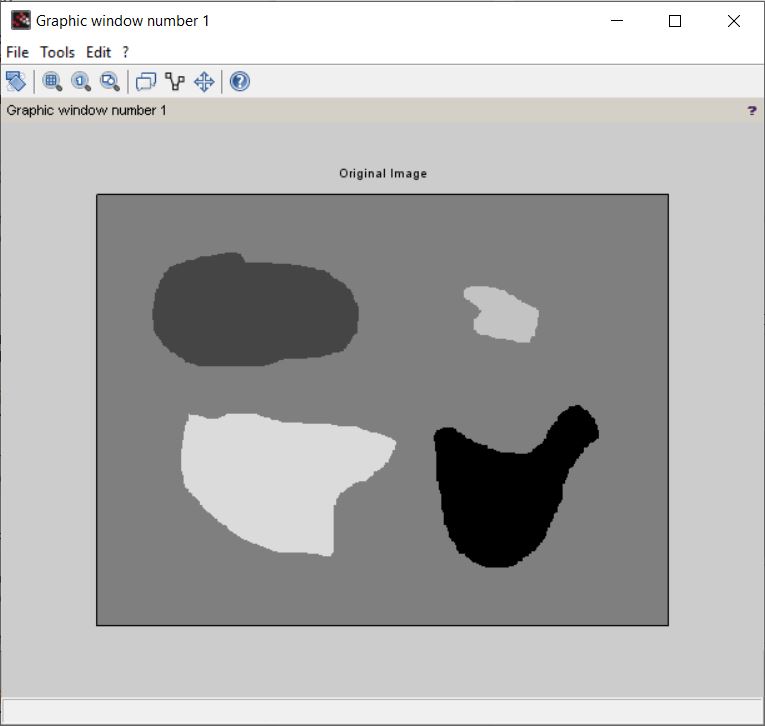
subplot(1,3,3),

imshow(uint8(255\*sobel/max(max(sobel))));

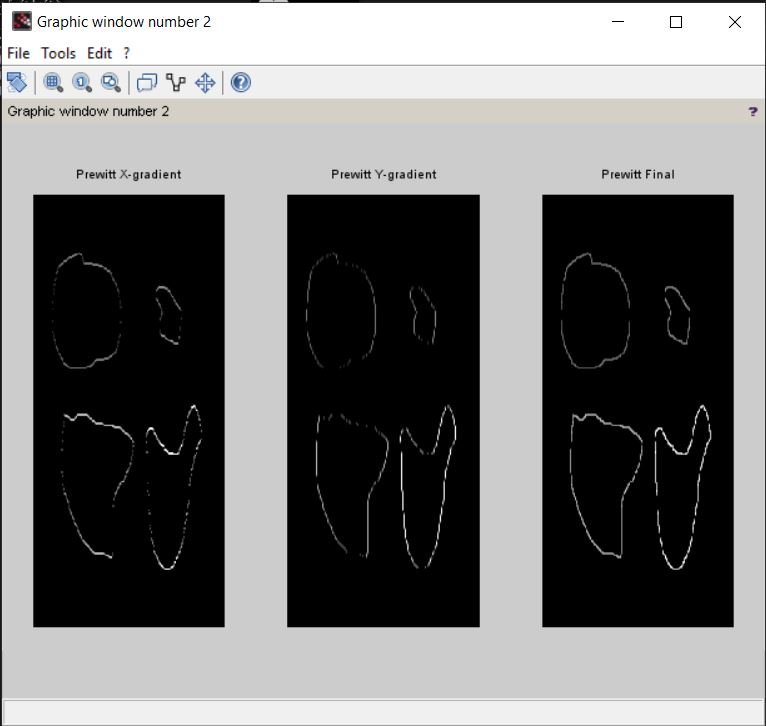
title('Sobel final');

**OUTPUT:**

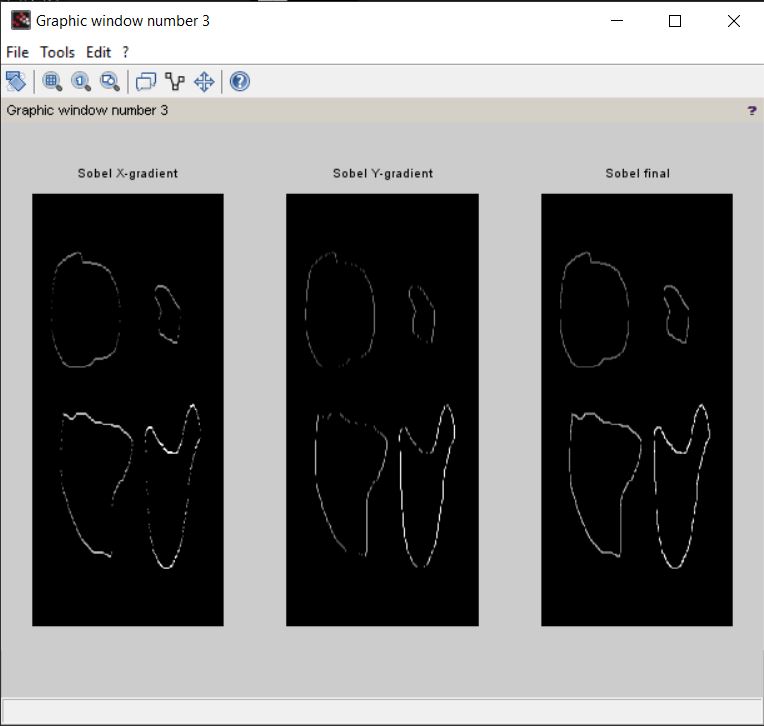
**Original Image:**

****

**Prewitt x and y gradient. And final Image:**

****

**Sobel X and Y gradient. And final Image**

****

**CONCLUSION**:

We studied and observed that:

1. Sobel edge detector is based on convolving the image with a small, separable, and integer valued filter in horizontal and vertical direction.
2. While, unlike the Sobel, prewitt’s edge detector does not place any emphasis on the pixels that are closer to the center of the mask.